AD-A100 432 BOLT BERANEK AND NEWMAN INC CAMBRIDGE MA F/G 9/2
BEYOND QUESTION-ANSWERING.(U)
MAY 81 P R COHEN, C R PERRAULT, J F ALLEN N00014-77-C-0378
UNCLASSIFIED BBN-4644 NL

1 OF 1
AD
A100432

END
DATE
FILMED
7-81
DTIC

**Bolt Beranek and Newman Inc.**

AD A100432

**Report No. 4644**

# Beyond Question-Answering

P. Cohen, C. Perrault, and J. Allen

JUN 2 2 1981

May 1981

Prepared for:
Advanced Research Projects Agency

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

# REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

**1. REPORT NUMBER**
BBN Report No. 4644

**2. GOVT ACCESSION NO.**
AD-A100 432

**3. RECIPIENT'S CATALOG NUMBER**

**4. TITLE (and Subtitle)**
BEYOND QUESTION-ANSWERING.
Technical Report
9

**5. TYPE OF REPORT & PERIOD COVERED**
Technical Report.

**6. PERFORMING ORG. REPORT NUMBER**
BBN Report No. 4644

**7. AUTHOR(s)**
10
P. R. Cohen, C. R. Perrault, J. F. Allen
Philip Raymond James

**8. CONTRACT OR GRANT NUMBER(s)**
15
N00014-77-C-0378,
NSF-IST80-12418

**9. PERFORMING ORGANIZATION NAME AND ADDRESS**
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02238

**10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS**
12

**11. CONTROLLING OFFICE NAME AND ADDRESS**
Office of Naval Research
Department of the Navy
Arlington, Virginia 22217
11

**12. REPORT DATE**
15 May 1981

**13. NUMBER OF PAGES**
61

**14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)**
14

**15. SECURITY CLASS. (of this report)**
Unclassified

**15a. DECLASSIFICATION/DOWNGRADING SCHEDULE**

**16. DISTRIBUTION STATEMENT (of this Report)**

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

**17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)**

**18. SUPPLEMENTARY NOTES**

**19. KEY WORDS (Continue on reverse side if necessary and identify by block number)**

question-answering, natural language understanding, artificial intelligence, plans, beliefs, speech acts

**20. ABSTRACT (Continue on reverse side if necessary and identify by block number)**


We demonstrate, using protocols of actual interactions with a question-answering system, that users of these systems expect to engage in a conversation whose coherence is manifested in the interpendence of their (often unstated) plans and goals with those of the system. Since these problems are even more obvious in other forms of natural-language understanding systems, such as task-oriented dialogue systems, techniques for engaging in (cont'd)


DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (cont'd)

question-answering conversation should be special cases of general conversational abilities. We characterize dimensions along which language understanding systems might differ and, based partly on this analysis, propose a new system architecture, centered around recognizing the user's plans and planning helpful responses, which can be applied to a number of possible application areas. To illustrate progress to date, we discuss two implemented systems, one operating in a simple question-answering framework, and the other in a decision support framework for which both graphic and linguistic means of communication are available.

Bolt Beranek and Newman Inc.
Report No. 4644

A

# Beyond Question-Answering

Philip R. Cohen
Bolt Beranek and Newman Inc.
Cambridge, MA

C. Raymond Perrault
Department of Computer Science
University of Toronto
Toronto Canada

James F. Allen
Department of Computer Science
University of Rochester
Rochester NY

This research was supported primarily by the Advanced Research Projects Agency of the Department of Defense and was monitored by ONR under Contract No. N00014-77-C-0378. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Projects Agency or the U.S. Government. The research was also supported in part by the National Science and Engineering Research Council of Canada under Operating Grant A9285 and by National Science Foundation Grant IST-8012418. This report will also appear in Strategies for Natural Language Processing, W. Lehnert and M. Ringle (Eds.), Lawrence Erlbaum, Inc., 1981.

TABLE OF CONTENTS

## ABSTRACT

We demonstrate, using protocols of actual interactions with a question-answering system, that users of these systems expect to engage in a conversation whose coherence is manifested in the interpendence of their (often unstated) plans and goals with those of the system. Since these problems are even more obvious in other forms of natural-language understanding systems, such as task-oriented dialogue systems, techniques for engaging in question-answering conversation should be special cases of general conversational abilities. We characterize dimensions along which language understanding systems might differ and, based partly on this analysis, propose a new system architecture, centered around recognizing the user's plans and planning helpful responses, which can be applied to a number of possible application areas. To illustrate progress to date, we discuss two implemented systems, one operating in a simple question-answering framework, and the other in a decision support framework for which both graphic and linguistic means of communication are available.

## ACKNOWLEDGEMENTS

This research was sponsored primarily by ONR Contract No. N00014-77-C-0378. Our thanks go to David Waltz, Harry Tennant, and Brad Goodman for providing and discussing the PLANES transcripts, and to Candy Sidner, and Bill Woods for their comments on the paper. Special thanks go to Norma Peterson for her invaluable help in preparing this paper.

## 1. INTRODUCTION

Judging from the number of implemented systems, one might conclude that the predominant application of natural language processing technology is question-answering (QA), usually from a highly structured data base. Recent systems have demonstrated enough robustness and coverage in their chosen subsets of natural language that users can accomplish significant work. While applauding the impressive results as a benchmark for future systems, we claim that interaction with current question-answering systems lacks naturalness, and that the structure of these systems imposes blinders on the development of other applications of natural language processing. This paper will both support these claims and propose a more general architecture for such systems, viewing question-answering as a special case of natural language dialogue.

We will demonstrate, using protocols of actual interactions with a question-answering system, that users of these systems expect more than just answers to isolated questions. They expect to engage in a conversation whose coherence is manifested in the interdependence of their often unstated plans and goals with those of the system.[1] They also expect the system to be able to

---

[1] The reader who is uncomfortable attributing mental states to machines should see [18, 41].

incorporate its own responses into analyses of their subsequent utterances. Moreover, they maintain these expectations even in the face of strong evidence that the system is not a competent conversationalist. We shall propose a program of research designed to develop some of the capabilities necessary for such interactions and will discuss progress to date.[2]

While some of the problems we identify might be solved by specific engineering methods, general techniques appropriate to other kinds of natural language systems, for example, decision support systems or task-oriented dialogue systems, are desirable. Ideally, techniques for engaging in question-answering conversation should be special cases of general conversational abilities. With generality in mind, we will characterize dimensions along which possible systems might differ and will situate various kinds of conversational systems in this multi-dimensional space. Based in part on the dimensional analysis, we will propose a new system architecture, centered around recognizing the user's plans and planning helpful responses, that can be applied to a number of possible application areas.

---
[2] Calls for similar programs of research can be found in [25, 29, 39].

Finally, to illustrate the progress to date, we will discuss two implemented prototype systems -- one operating in a simple question-answering framework, and the other in a decision support framework for which both graphic and linguistic means of communication are available.

## 2. THE TRANSCRIPTS

Two sets of data have been particularly useful. First, we have been fortunate to receive access to voluminous protocols of teletype interactions with the PLANES system, a natural language question-answering system that deals with a relational data base of aircraft flight and maintenance records.[3] The architecture of PLANES is described by Waltz [63] and its linguistic and conceptual coverage are presented by Tennant [61]. To test PLANES, users were asked to fill out a table, histogram, or graph. The PLANES system translates each query from natural language into an expression in a formal query language that is then evaluated against the data base. In response, the user is given an English paraphrase of his query and, if the system's analysis is accepted, a tabular output or the result of some simple computation (e.g., SUM, AVERAGE) over selected rows or columns of the retrieved tables.

We have also received protocols of users interacting with a "simulated" PLANES -- one in which the natural language system

[3] PLANES is a worthy exemplar of current natural language technology. The problems we point out are by no means unique to it. Other systems (including LADDER [27, 49], LUNAR [68], REL [62], RENDEZVOUS [14, 13],and ROBOT [26]) are similar with respect to the capabilities we are discussing, though they may differ in other areas.

was replaced by Harry Tennant [61].[4] The "Simulated" system thus was intended to demonstrate capabilities of an ideal PLANES.

## 2.1 The PLANES Transcripts.

The vast majority of interactions with PLANES were of the expected sort -- straightforward queries expressed in natural language, often with a complex quantificational structure. The structure of the queries, naturally, depended to a great extent on the problems the users were asked to solve. We shall not examine these since their analysis and related difficulties are well-known.

Two kinds of phenomena arise in the protocols that lead us to claim users were trying to engage in dialogue. First, they expect the system to infer and respond to their apparent but unstated goals. Secondly, users' subsequent utterances depend on the system's responses. The following dialogue illustrates these

---

[4] For similar studies, see also [19, 24, 37, 60]

two points. S is the system, and U the user.[5]

D1-1 U: Which A7's had normu of greater than 100 hours between april and may 1973

2 S:

| ACTDATE | SERIAL# | PLNTYP |
|---|---|---|
| 304 | 4 | AAF |
| 305 | 5 | AAF |
| 305 | 9 | AAF |
| . | . | . |
| . | . | . |

3 U: No, How many normu hours greater than 100?

4 S: I could not understand the word NO, will skip it.

5 S: ((COUNT BUSER) (SUM NORMUNS)) = (4,644)

6 U: Which four?

7 S: [S translates utterance to an incorrect query. User does not allow system to continue]

8 U: Which 4 planes had normu's greater than 100 hours?

---

[5] These abbreviations recur in the dialogues:

A7 — a type of aircraft, also symbolized as "AAF".

NORMU — number of hours an aircraft was Not Operationally Ready due to Unscheduled Maintenance

ACTDATE — Date (Year/Month) aircraft was NOR

SERIAL# (also BUSER) — unique identification number of an aircraft.

PLNTYPE — type of aircraft (e.g., A7, F4).

9 S: [program fails]

In utterance D1-1, the user issued a typical "query" of the kind PLANES is equipped to analyze. Though the system replied to D1-1 with what, on the face of it, seemed to be a correct answer,[6] it did not tell the user what he wanted. The user wanted another column of output indicating the number of NORMU hours for each plane. He tried to indicate in D1-3 that the system's response was somehow unsatisfactory, by typing "No". The subsequent "how many" question in D1-3 was intended to be interpreted as a request to <u>modify</u> the previous answer to include the currently desired information. However, PLANES ignored the "No", interpreted the remainder as an isolated question, and correctly answered it.

The second source of evidence to support our claim that users expect to engage in conversation, is the dependence of their utterances on <u>the</u> <u>system's</u> responses. In particular, references to table entries, or to the name of a function and its value, are often incorporated into the users' subsequent questions. For instance, in D1-5, PLANES printed the value of the function "number of planes" as 4. The user's next utterance

---

[6] Actually, the answer was incorrect. The system considered the flight hours and not the NORMU hours greater than 100.

asked for further information about those four planes. The system, of course, simply evaluated a function and printed its result, without recording that it had communicated the existence of four planes. The same problem recurs in the following fragment.

D2-1 U: What parts were repaired on buser 48 between may 16 1969 and may 17 1969.

2 S:

PARTNO
522-0885-005

3 U: What was wrong with it

Thus, whether the response is tabular, LISP-like or in natural language, "discourse entities" [64] should be made available for subsequent reference.

Assuming a system can keep track of what it has communicated, should it always do so? We think so -- users responses, as illustrated in the next PLANES interaction, can depend on even the most "canned" phrases.

D3-1 U: Give me the planes that flew more than 10 flights in 1970, according to the number of flights.

2 S: (OUTPUT SCHEDULED, THERE WERE 60 ITEMS)

3 U: How many was that? How many flights did they each fly?

The system's "message" indicates that it has deferred supplying the answer since too much information has been retrieved. The user is clearly dissatisfied with the response and requests further information. He is indicating, as he did in D1-3, that the system's response does not meet his goals in asking the previous question.

It should be clear that these PLANES interactions have many characteristics commonly associated with conversational interaction. Users expect the system to be responsive to their (often unstated) desires. They also expect the system to be able to interpret <u>their</u> utterances in light of <u>its</u> responses.

## 2.2 Protocols with a simulated PLANES

To show how humans might wish to converse about data bases, consider the next dialogue that took place between a user and a simulated PLANES (S-PLANES). The PLANES and S-PLANES protocols were markedly different. Obviously, some of the variance can be attributed to the different "systems" involved (S-PLANES included a person). But, as Tennant points out, part of the difference arises from the dialogue situations -- the users of PLANES were asked to fill out a table, while the users of S-PLANES were asked to solve a problem (discover why a higher percentage of aircraft in 1971 were "not operationally ready"). The "table-filling"

situation undoubtedly led to uniformity in the kinds of questions asked, while the problem-solving situation, coupled with the users' knowledge that Tennant knew the problem to be solved (and its answer), led some users to state the solution to the problem or to ask S-PLANES for advice. We shall look again for dependencies of the "system's" responses on a model of the user's goals, and for linguistic and pragmatic[7] dependencies of the user's utterances on the system's responses.

D4-1 U: How many cases of FOD were experienced

2 S: (THAT QUERY WOULD TAKE ABOUT 20 MINUTES TO COMPUTE) (IT CAN BE USEFULLY LIMITED BY CONSIDERING SPECIFIC AIRCRAFT)

3 U: Consider aircraft #27

4 S: NIL

5 U: How far from a supply base is PUC 38 and 306

---

[7] More abbreviations:

| | |
|---|---|
| PUC | Permanent Unit Code -- codes for permanent bases of various aircraft |
| AWM | AWaiting Maintenance times |
| A/C | AirCraft |
| NOR | Not Operationally Ready |
| FPC | Flight Purpose Code |
| BUSER | BUreau SERial # -- a unique identification number for an aircraft |
| FOD | Foreign Object Damage |

6 S: (I DO NOT HAVE THAT INFORMATION)

7 U: Is there a record of AWM times for these PUC's

8 S: (YES)

9 U: Does it have to be obtained using A/C numbers or can I get a complete listing?

10 S: (IT WOULD TAKE ABOUT 20 MINUTES TO DO A COMPLETE LISTING. THERE ARE FEWER PLANES IN 1970, AND 1973, SO THESE WOULD NOT TAKE SO LONG)

11 U: Consider A/C #46

12 S: (THERE IS NO AIRCRAFT 46, WOULD 34 BE OK?)

13 U: Yes

14 S:

| UNIT CODE | AWM-PER-PUC |
|---|---|
| 10000038 | 3389.5006 |

15 U: Was the A/C away from it's assigned PUC at the time it became NOR?

16 S:

| BUSER | AWAY |
|---|---|
| 34 | A |

17 U: What was its FPC?

18 S:

| BUSER | FPC |
|---|---|
| 34 | 1A4 |
| 34 | 4A1 |

19 U: I don't understand these numbers

20 S: [system explains coding scheme]

Although this is clearly a conversation, the factors underlying the coherence even of utterances D4-1 to D4-3 are far from obvious. In isolation, "Consider aircraft #27" is extremely vague,[8] yet it becomes precise in light of the "system's" response D4-2 -- namely, as a suggestion to try answering D4-1 narrowed to aircraft #27.

The same problem of unsatisfied goals occurs in D4-7 where the user asked whether there was a record of the length of time aircraft were awaiting maintenance at two bases. The user, of course, wanted the AWM times and believed Tennant knew that.[9] Tennant responded "literally", giving a positive answer to the yes/no question. By not responding to the unstated but obviously related goal of getting the system to display the AWM times, Tennant communicated that he was aware that the user's goal was unfulfilled. Utterance D4-9 shows that the user too had realized there was some reason why the system was not addressing the intent of his question. In hot pursuit of the AWM times, the participants engaged in a long subdialogue about how to obtain a variant of the data implicitly requested in D4-7. Finally, in

---

[8] Consider the situation of trying to sell someone a used airplane, and uttering D4-3.

[9] Tennant confirms that this is the case.

D4-11, the user gave his now familiar request to "Consider" a particular aircraft. This time, there were still more difficulties, and another subdialogue (D4-12, D4-13) took place to recast D4-11 to specify an existing aircraft. When the system produced a response in D4-14, it was actually responding to the user's goal first addressed (though not literally stated) in D4-7. Finally, in D4-19, the user requested an explanation by stating his problem.

We believe systems can be built to partake in similar dialogues. Since it appears users of question-answering systems expect those systems to analyze and respond to (certain of) their goals, we examine now how these goals can be uncovered.

## 2.3 Non-literal uses of language

It is well known that people do not say precisely what they mean, even to question-answering systems. Rather, they fully expect their hearers to infer many of the intentions that motivate their utterances.

Speakers can have many different intentions behind even the most simple of utterances. For example, a user of a natural language front end to a data base system may have many different goals for stating "The flight number is 732" -- she may be simply informing the system of some fact, or correcting a previous

system response, or asking the system to check its information. The utterance may even be "part of" the making of a request, as in "I need to know the departure time for the flight to Indianapolis. The flight number is 732." Furthermore, speakers can have multiple simultaneous intentions in making an utterance. For example, the following utterances typed to PLANES exhibit both a "literal" and a "closely related" intention:

1 I request the number of flight hours for buser 4 on June 26, 1973.

2 I need to know the number of flight hours flown during June 1972 for aircraft with number 13.

3 Want number of flight hours flown by number 13 during June, 1972.

4 Find the number of F4 aircraft that were NOR in July 1972.

5 Was any work performed on Plane 3 from june 1 to 7, 1973.

Utterance 1 is a performative (c.f. Austin [4]) -- it is the performance of a request and not a statement of a request.[10] "I need to know..." and "Want..." are just statements of the user's goal. The system is not expected to respond simply with "I understand", or "OK", but rather to do something to satisfy that goal. Similarly, in 4 the system is not only expected to

---

[10] Actually, the utterance in the transcripts was an elliptical performative "Request the number of flight hours..." We return to this example in section 3.

retrieve information, but also to inform the user.[11] In 5, the speaker wants to know what work was performed (if any) and not simply whether any work was performed. Utterances such as these that nominally convey one intention but are being used to communicate another are called <u>indirect speech acts</u> [56].

While some intentions are closely related to the utterance form, others are quite far removed (e.g., "Consider A/C #27" or "I don´t understand these numbers"). In all these cases, however, the system has to be sensitive to what was <u>literally</u> said since, for example, it might need to respond negatively to 5.

To complicate matters, occasionally <u>only</u> the literal interpretation is intended. "Can I get a complete listing" could be used to request a listing, but in D4-9, repeated below, it isn´t.

> 6 Does it have to be obtained using A/C numbers or can I get a complete listing?

A system must know when a form is used with just its "literal" intention, and when it should infer other related intentions. Moreover, it must know when to stop -- when various possible

---

[11] Speakers of "find..." requests in task-oriented dialogues [24, 17] do not necessarily expect to be informed of what was found.

intentions should not be attributed to the user. The problem for a conversation system, then, is to infer those intentions of the user that it was intended to infer.

## 2.4 Being Helpful

One striking aspect of the S-PLANES protocols is the extent to which Tennant discovers difficulties with the user's queries and suggests alternative means to achieve the same or related goals. For instance, in D4-2 and D4-10, Tennant notices that an answer to the user's question will be too expensive to compute. Instead of simply stating that fact, he goes on to state how the query might be modified to be more efficient. Similarly, in D4-12, he notices an erroneous presupposition, reports that fact, and suggests an alternative. Kaplan [32] presents a partial solution to this problem for data base retrieval queries. However, we claim presupposition correction is a specific case of a more general failure of someone's plans. The model of cooperative conversation proposed in section 4 will show how a machine can detect plan failure, and suggest alternative paths to achieve the same, or a related goal.

## 2.5 Clarification Dialogues.

Quite frequently, the user must communicate his intentions to the system through a "negotiating" process. This is reflected in what can be called clarification subdialogues. Some of the simplest instances in Codd's RENDEZVOUS system [14, 13], and in PLANES occur where the system presents the user with a supposedly unambiguous reformulation, in English, of his English query. The user is then asked to either confirm the reformulation or to modify it through a simple editor. The process iterates until the user is satisfied or withdraws the query altogether.

More complex dialogues result if the system detects ambiguities in the input. Winograd [66] and Codd handled these by asking the user to choose among the interpretations. However, if the entire interaction is to be done in natural language, the system must be able to formulate a question whose answer can allow it to discriminate between the original interpretations.

It is also possible for the user to reformulate his original utterance ignoring the clarification question. This requires the system to recognize that the clarification question is not being answered. Consider the following fragment from an S-PLANES transcript.

D5-1 U: Print the NOR times for aircraft in 1971 and 1970.

2 S: Do you want the totals for all aircraft, or averages, or totals for each?

3 U: Totals for each aircraft, by year and serial number.

The question D5-2, for example, could have been followed by 7.

7 Print the NOR times for each aircraft.

Another source of clarification dialogues is the conflict of stated intentions with standing ones. This covers the cases in the S-PLANES dialogues where the system finds that the resources necessary to answer a question may be greater than the user thought.

## 2.6 Summary

We have given examples of several problems with current question-answering systems. First, their users expect them to react to unstated goals. This is evidenced by their rejection of the system's interpretation of their intentions and their attempts to make their intentions understood without completely restating their queries. It is also demonstrated in their use of indirect speech acts. Second, users may make complicated requests in several utterances, each one providing more detail to the previous ones. The final form of a request is sometimes the result of a "negotiation" with the system about how things can be

done. Third, the user expects the system to be aware of the user's reference failures, and more generally of the failures of his presuppositions, and to ensure that the user is not mislead by the incorrect assumptions. Finally, the system should expect that the user's utterances will depend on the system's. This paper, and our research program, concentrates on how the user's intent can be inferred. Before presenting a framework in which to couch our proposed solution, we develop means to compare language systems.

## 3. COMPARING LANGUAGE UNDERSTANDING SYSTEMS

Most question-answering systems have three main constituents: an analyzer translates the user's utterances into expressions in an unambiguous query language; a retrieval component fetches from the data base a set of records according to the query; and a generator simply lists the extracted records, information they contain as a natural language utterance. Control then returns to the analyzer to process the next query. This simple view cannot be maintained for systems that properly handle the problems outlined in the previous section. We will sketch a different picture, and indicate steps that have already been taken to implement parts of it.

Before suggesting these changes, we discuss some relations between the problems by presenting several dimensions along which one can compare the capabilities of language understanding systems in general. We suggest that the problems can be solved by extending question-answering systems along these dimensions. The dimensions are: versatility, discrimination, context-dependence, single-mindedness, and helpfulness.

### 3.1 Versatility and Discrimination

The user sees the system he is working with as being able to perform a range of functions, both linguistic and non-linguistic.

In some systems the range is highly restricted, e.g. answering questions from a static database or giving commands to a robot. In other systems it is broader,[12] e.g. question-answering and (simulated) hand movements (SHRDLU [66]), answering and asking questions (LUNAR [68], RENDEZVOUS [13], LADDER [27, 49], REL [62], ROBOT [26]), asking, answering, and requesting (TDUS [48]), asking, answering, and responding to requests (HWIM [67]). We will call the range of functions a system can perform its <u>versatility</u>.

The user of a language understanding system intends his utterances (and maybe even some of his other actions) to have some effect on the system's behavior. Let the <u>discrimination</u> of a system be the degree to which it can recognize in the user's

---

[12] The lists of systems given in this paper are not meant to be exhaustive. Our apologies if your favorite is missing.

actions the intentions the user wants it to conform to.[13] For example, the system might have to recognize that the user intends it to provide information, accept, correct, or check information, make physical movements, etc. If the user is to control the system, the greater the system's versatility, the greater the repertoire of messages the user needs to be able to send it. This in turn makes the system's understanding problem more difficult as it must be more discriminating in its analysis of

---

[13] Two kinds of discrimination can be distinguished: functional discrimination is the ability to recognize functions to be performed, and content discrimination is the ability to distinguish the "arguments" to those functions. For example, a system might distinguish between questions and assertions (functional discrimination), while a system with high content discrimination might also recognize questions of high complexity, e.g. ones containing boolean operators, quantifiers, etc. Previous analyses of the performance of language understanding systems limited themselves to question-answering systems, and proposed scales which we consider for the purposes of this paper to be subsumed by content discrimination. Woods [68] writes:

> A system is logically complete if there is a way to express any request which it is logically possible to answer from the data base. The scale of fluency measures the degree to which virtually any way of expressing a request is acceptable.

Tennant [61] uses the terms conceptual and linguistic completeness for completeness and fluency, respectively, and introduces conceptual and linguistic coverage to measure the user's expectations about what queries he should be able to make of the system. This distinction between system capabilities and user expectations about them should be extended to functional discrimination. See also [58].

the user's utterances. For example, if a system that can only answer questions is told

8 There are 3 flights a day from Boston to Toronto.

it must interpret the utterance as a yes/no question, or reject it altogether. A system that can both answer questions and update its data base, acting only on the basis of the syntax and semantics of the sentence, would probably interpret it as an assertion, although in some contexts the user might intend it as a question.

Most question-answering systems would try to analyze 9 as an imperative, and would not know what to do with it. Some would then ignore the verb altogether, and treat the remaining noun phrase "the number of flight hours..." as a request for the system to tell the user the number of flight hours. This would turn out to give the right result if the user meant 9 as an elliptical form of 10.

9 Request the number of flight hours for buser 4 on June 26 1973.

10 I request the number of flights for buser 4 on June 26 1973.

In some circumstances, this interpretation would be wrong. Consider a system that acted as the hub of a group of users and could pass assertions and requests from one user to another. It could interpret 9 as a request to the system to request another user to tell him the number of flight hours.

### 3.2 Context-dependence

In its simplest interpretation, the analyze-retrieve-generate scheme assumes that what the system does after it has analyzed an utterance depends only on that utterance. There are at least three ways in which one may wish to relax that assumption, and they are the basis for the next three dimensions.

First of all, and most obviously, the behavior of the system after an utterance may depend on the previous utterances. This dimension we call context-dependence. Some systems do not depend on the context at all. For example, a data base question-answering system in which the input language is an unambiguous query language is context-independent since the order in which a set of questions is asked has no effect on the set of answers. PLANES and virtually all other question-answering systems make use of some form of context to complete the content of an utterance, in particular to determine the reference of pronouns, and to recover missing verb phrases.

Along with many others, we take "context" to mean, roughly, the shared beliefs available to the system and the user as a result of the discourse itself, the medium of communication, the physical setting the participants can perceive visually, and general knowledge assumed by the participants. Intentions of both participants may also be shared.

Before considering shared beliefs, a few comments on simple beliefs are in order. First, we assume that the system has no direct access to the user's beliefs, and thus can only have beliefs about the user's beliefs. Also, in general, what the system believes can be different from what the system believes the user believes, and from what the system believes the user believes the system believes, and so on.

How many of these distinctions must a language understanding system be able to make? This depends on its versatility and discrimination. In the simplest context-independent question-answering systems, repeating a question elicits the same answer each time. The system has no history of what it has already told the user and cannot avoid repetition. It acts as if it were not distinguishing its own beliefs (the data base) from those of the user (or rather from its beliefs about the user's beliefs). If a system is expected to not tell the user what he already knows, or to correct the user's false beliefs, then it must be able to make this distinction. Any system versatile enough to make and defend assertions must therefore distinguish at least three *levels of belief*: what it believes, what it believes the user believes, and what it believes the user believes about what it believes.

Some version of shared belief is necessary to the correct understanding and generation of definite descriptions. For example, suppose that the system and the user have expressed different views about the referent of a definite description, so that the system believes that the referent of "the captain of the Enterprise" is Spock while believing that the user believes him to be Kirk. Having publicly expressed its belief, the system is also justified in believing the user believes it to believe that Spock is the captain.[14] In the circumstances, "the captain of the Enterprise" cannot be used reliably by either system or user to refer to either Spock or Kirk. Any strategy to, say, generate definite descriptions that only uses a single, fixed, level of belief will not be sensitive to the disagreement, and thus cannot be prevented from generating "the captain of the Enterprise" to refer to one of Spock or Kirk. Similar problems arise with understanding definite descriptions.

Understanding and generating descriptions correctly therefore requires the agreement of at least two levels of belief. Could these be what the system believes and what the system believes the user believes? We claim not. Suppose that

---

[14] Schank and Abelson's [50] use of MTRANS to model the act of asserting does not capture these distinctions.

at first system and user agreed that Kirk was the captain. Then suppose that the system found out through direct, private access to the Enterprise that Kirk had been replaced by Spock. The system would therefore believe that Spock was the captain, while believing that the user believed that Kirk was. The user's utterance of "the captain of the Enterprise" still clearly identifies Kirk, and should be understood as such by the system. But, this cannot be done if referent identification depends on "P is shared by S and U" being defined as S believes P and S believes U believes P.

The next most obvious version of sharing, agreement of what S believes U believes with what S believes U believes S believes, works in this case and is adequate for many purposes. A more comprehensive but, it turns out, no more onerous account of the shared belief that P can be based on the <u>mutual belief that P</u>: a predicate equivalent to an infinite conjunction of beliefs of the form

> 11 S believes P <u>and</u> S believes H believes P <u>and</u> S believes H believes S believes P ...

A related notion was first introduced by Lewis [36] and Schiffer [51]. Clark and Marshall [12] discuss the acquisition of mutual beliefs; they and Perrault and Cohen [46] show how it is related to the use of referring expressions; and Cohen [16] presents a data structure that allows a finite representation.

Anaphora and reference has long been of interest to computational linguists. Webber [64] shows how descriptions can be used to evoke new entities in the discourse. Grosz [23, 24], Sidner [59], and Reichman [47] discuss how task structure, syntax, and topic can restrict which of those entities the speaker intended to refer to using a pronoun or a definite description. The relation between the work on discourse entities and focus, and that on shared beliefs, however, remains to be established.

Anaphora resolution is only one of the problems requiring the use of discourse context. Another is the understanding of intentions communicated in several utterances or turns. This is necessary if the user is to be able to <u>state</u> general constraints on how his utterances are to be interpreted. For example, if 12 had preceded 13 (repeated here from D4-1), then in replying that it would take 20 minutes to compute an answer, the system would merely be complying with the user's stated intentions.

12 Tell me if I ask you to do something taking more than 20 minutes.

13 How many cases of FOD were experienced?

The more discriminating and context-dependent the system, the more the user can "fine-tune" its responses to his stated intentions.

### 3.3 Single-mindedness and Helpfulness

System designers tend to think of their systems as doing what the user wants, no more, no less. But what intentions was S-PLANES/Tennant considering in replying as he did to 13? He could have been assuming that the user did not want long computations to be performed without confirmation, although the user never explicitly stated this. He could also have been simply refusing, on his own authority to expend the necessary resources. This is one example where the system may not be completely single-minded, that is, responsive only to the intentions of the user. Another case would be if the system refused the user access to data protected by another user. The system will always have to make decisions based on intentions not explicitly communicated by the user.

Even a single-minded and context-dependent system can be irritating. For example, if the user believes the system has sufficient information to know that an action the user is about to attempt to perform is likely to fail, then the user will expect the system to at least inform him of the situation. A system that can predict the failure of a future action of the user's, and respond appropriately, we call helpful. Consider the following example (repeated here from D4-7):

14 Is there a record of AWM times for these PUCs?

If the system knows that there is such a record, but does not have access to it, and believes that the user wants to see it, a reply of "Yes" is undesirable since it leads directly to:

15 U: Well, give it to me.

16 S: I don't have it.

A reply of:

17 Yes, but you'll have to do such and such to get it.

is closer to what the user intended. Kaplan's presumption failure correction mechanism is yet another example.

Thus one is forced to abandon the simple view that only the meaning of the user's last utterance (or the intentions conveyed by it alone) is sufficient to determine the subsequent actions of the system. As a consequence, the retrieval component and the generator must be replaced by a process by which the system determines its subsequent actions based on the user's intentions, implicit or expressed over time, and possibly the intentions of others.

### 3.4 Summary

Versatility, discrimination, context-dependence, single-mindedness, and helpfulness are independent dimensions of system behavior in that one can conceive of language understanding systems with high values for some and low values

for others. Systems tend to be designed with versatility and discrimination of the same order; otherwise, a system could understand intentions it couldn't satisfy and vice-versa. Although the dimensions may be independent, the solutions to some of the problems raised by the transcripts, in particular clarification dialogues and indirect speech acts, require extending question-answering systems along several of them.

For a system to engage in natural language clarification dialogues, it must be able to formulate questions whose answers will allow it to choose among the original interpretations, or reject them, altogether. This requires more versatility than the simple question-answering systems have. For example, being able to recognize that an answer to a clarification question in fact is a rejection of any of the alternatives presented in the question requires more discrimination than any current systems have. In all these cases, the system's behavior depends on the context established through several utterances.

Similarly, being able to recognize indirect speech acts correctly (i.e. being able to attribute to the user intentions not literally associated with the form used) requires more discrimination than current question-answering systems have. This discrimination relies on context and on knowledge of the process by which agents cooperatively adopt intentions of others

as their own. An utterance can be used indirectly to convey intention B if it could be used literally to convey intention A, and if cooperative behavior by the user would lead him to infer that the speaker intended B as well as A.

The remainder of the paper proposes and justifies an approach to the discrimination of the user's intentions and to the generation of helpful behavior. It is independent of the particular kind of language understanding system being considered. It identifies intentions with plans, and views utterances as planned by speakers to achieve effects on hearers.

## 4. PLANS AND COMMUNICATIVE ACTS

Philosophers of language, in particular Austin [4] and Searle [55], have suggested that all utterances be viewed as resulting from purposeful actions. English contains a large vocabulary of terms that label these communicative, or speech acts, e.g. request, demand, assert. These terms have been used liberally in section 2 to describe the user's intentions in the sample dialogues. As suggested by Bruce [8, 9] and Schmidt [53, 54], we propose that language understanding systems be able to both make such judgements and perform such actions. Neither is a simple problem since there is no direct mapping of utterance form to the action it is being used to perform. Rather, the system must engage in a process of reasoning about how an utterance is being used (i.e., what are the user's intentions), what communication actions it should perform, and how they should be performed.[15]

One benefit of viewing utterances as actions is that we can take advantage of work on reasoning about actions, both formal (McCarthy and Hayes [40], Moore [42]) and informal (GPS [44], STRIPS [20], and NOAH [49]). Most of the informal literature

---

[15] Compatible proposals are made in [6, 35, 38].

is concerned with planning, or what we will call plan construction, the process of finding a (complex) action (or action sequence) that will transform a given state of the world into one satisfying a given goal.[16] Plan construction algorithms allow an agent to examine the consequences of sequences of future actions before executing any of them, i.e. before making any changes in the outside world. Some of the planned actions can be communication actions, and these lead to changes in the states (beliefs, intentions) of other agents [8, 15, 30, 53].

Just as it is useful for an agent to be able to consider future actions without actually doing them, it is also useful to observe actions performed by some agent, and predict what subsequent actions he intends should be performed, either by him or by someone else. The process of inferring the plan an agent may be following is called plan recognition. An observer's recognition of an agent's plan is performed on the basis of beliefs about: the agent's beliefs, conditions that are likely to be true at the end of an action, other actions that are enabled by those conditions, and likely plans and goals of that agent.

---

[16] We shall consider all the actions, and states-of-affairs relating them, in an agent's plan as being, on balance, wanted by the agent.

The object of this section is to show how plan construction and plan recognition can be used to provide the basis for solutions to some of the intention discrimination problems identified in the transcripts.

What distinguishes acts of communication from the others (as pointed out by Grice [22] and by Schiffer [51]) is that not only are they performed with the intention that they should have some effect on the hearer(s) (e.g. that the hearer should believe something, or want something) but also that they be performed with the intention that these effects should come about in a particular way, to wit, through the hearer's recognizing that the speaker intends the hearer to believe that the speaker is trying to achieve these effects. The system, therefore, cannot simply infer and act on what the user wants (as if it were observing the user through a keyhole), but must infer and act on what the user wants it to "think" that he wants. This last inference process, termed <u>intended plan recognition</u>, relies on shared beliefs and is the means by which acts of (Gricean) communication are performed. In contrast, being helpful involves keyhole plan recognition.

Must a system embody such seemingly complex reasoning? Why not have it reason only with its own wants and goals, ignoring the user's? If it were somehow given a goal by the user (a computational "injection"), it might plan a course of action that

the user did not in fact want. At a minimum, one would like a planning system to at least verify that the user would want its planned action(s).[17] Therefore, at a minimum, the system needs to reason about the user's wants.

Why not then reason only about the user's wants? Why should the system maintain wants of its own -- i.e., why shouldn't it be single-minded? If a system is not to be required to do everything a user wants, that system needs to maintain the distinction between its own wants and wants it attributes to the user. For example, one might not want an automated banking system to attempt to satisfy the want expressed by "Make me a millionaire."

The intended versatility of a system thus can justify having it distinguish between its own wants and those of the user. Now, assume the system can distinguish communicative from non-communicative acts (as might be needed for a natural language graphics system that also allows standard keyboard control of some display functions). We will sketch a minimal process for reasoning about the user's wants and show that when it is applied to suitably defined communicative acts, it leads to a complex of beliefs and desires necessary for intended plan recognition.

---

[17] The verification might be done via the planning of a question. More on this soon.

Assume the natural language system "observes" the user perform a non-communicative act, e.g., moving the "mouse" on a tablet. The system infers (or assumes) the act was intentional -- the user wanted to do it. It is then reasonable for the system to infer that the user wanted the typical effect of that action (that the cursor be at a different location on the screen). Furthermore, the system may infer the user wanted that effect because he believed it would allow him to perform some other action, such as moving an entity on the screen. This keyhole plan recognition process, if successful, yields a plan attributed to the user. Schmidt, Sridharan, and Goodson [52], and Wilensky [65] have developed such plan recognition algorithms.

Even in a situation where two agents are not attempting to communicate, it is possible for one to assist the other by observing his actions, inferring his plans, detecting obstacles in these plans, and attempting to overcome them. The obstacle detection phase can be thought of as a verification that the inferred plans will in fact achieve the inferred goals. If they do not (i.e. if the observer's knowledge of the world and of the actions to change it is different from what the observer believes to be the agent's knowledge) then the observer should be able to adopt as his own the agent's soon-to-fail goal. Once it has been adopted, the new goal can be solved by the observer's plan

construction mechanism. Genesereth [21] and Allen [2] have shown that a system that has inferred a plan for the user can be helpful by ensuring that the plan succeeds. Discussions of the way plans of different agents can interact can be found in [7, 11].

What happens if the user types an utterance? The system again "observes" an action, e.g., the uttering of an imperative, interrogative, or declarative sentence, and infers the user wanted the typical effect of that action. What are the typical effects of such acts?

A plausible effect would be, for an imperative, the hearer's believing the speaker wants the hearer to do some act A. Thus, the system would believe that user wanted it to do A (abbreviated "SBUW (Do S A)"). But having assumed the act to be intentional, the system would also believe the user wanted the effect of the imperative. Therefore, it would have inferred a proposition of the form: SBUW(SBUW(Do S A)) -- i.e., it would believe that the user wanted it to think he wanted it to do A. This proposition is the starting point for the process of _intended plan recognition_. Further inferences of the form SBUW(SBUW(A)) --> SBUW(SBUW(B)) allow the system to infer other goals the user wanted the system to think he had. Any such goals inferred during intended plan recognition, are now goals the system was _supposed_ to attribute

to the user and hence (according to Grice) have been communicated. The discovery of such goals is the heart of indirect speech act recognition [45].

The problem of controlling inferences arises for plan recognition, as it does with any inference process. Allen and Perrault [1] show how plan recognition should terminate successfully when a line of inference connects with an expected goal of the user [18] The expected goals may be specific to the user, or depend on his membership in a class of users with typical behavior patterns.

A special heuristic is useful to control <u>intended</u> plan recognition inferences. It is based on the assumption that the speaker is a rational agent, and thus only intends inferences to be drawn if they can be drawn unambiguously. The heuristic therefore terminates intended inference chains that lead to mutually exclusive alternatives, for which the hearer has no reason to select one over the others. Of course, the success of this heuristic depends on the accuracy of the models the speaker and hearer maintain of each other, a not unreasonable condition.

---

[18] See [10, 35, 47, 50, 52, 65] for compatible uses of expected goals.

We have argued that intended plan recognition arises naturally from a keyhole plan recognition process that requires:

1. observing an utterance of a sentence,
2. assuming the agent wanted to do it,
3. inferring that the agent wanted the typical effect of the act,
4. characterizing the effects of the uttering of sentences to be hearer beliefs about the speaker's wants.

Steps 1), 2), and 3) have independent motivation, while step 4) was justified intuitively. Could not the proposition produced by an utterance, say an imperative, be simpler? For example could not the effect of uttering an imperative be that the hearer wants the act A? Ultimately, which proposition is made true by uttering sentences of a particular form is a decision of the system designer, but there is good reason not to have imperatives, for example, <u>always</u> cause the system as hearer to have new desires. For example, one might not want a system, told to change the user's salary, to come to have that as a goal that it would plan to achieve. Therefore, to "insulate" the system and allow it to reason about the user's desires, the effect is represented by "Hearer believes speaker wants Act". A similar argument can be made for definitions of acts of uttering declaratives and interrogatives.

At this point, then, all four steps have been justified. The system performs intended plan recognition as a by-product of a process of reasoning about the intentions underlying the user's actions that is applied to linguistic acts.

To illustrate this process, assume that the user tells the system "Do you know where the Enterprise is?". From the syntax and semantics of the question the system recognizes that the user intends it to believe that the user wants to know whether the system knows where the Enterprise is. From this, the system can: infer that the user in fact wants to know whether the system knows where the Enterprise is, then adopt the user's knowing whether the system knows as a goal, then satisfy the goal by telling the user, whether it knows or not. The system would then have complied with the user's literally stated intention.

But if the answer turned out to be "Yes", the system would be in most cases less than helpful, since the user would probably be expecting the system to tell him where the Enterprise is. As pointed out in section 2.3, this is not always the case.

Having inferred that the user wants the system to recognize whether the system knows where the Enterprise is, the system can infer that the user intends the system to recognize both that the user wants to know where the Enterprise is, and that the system should tell him. Thus, the user uttering "Do you know where the

Enterprise is?" can, in some circumstances, convey the intentions which could have been explicitly communicated with "Where is the Enterprise?". In others, he can be conveying only the intentions associated with the yes/no question. The system's intended plan recognition process and the knowledge it has of the user and the world allow it to choose among the interpretations.

## 4.1 Summary

We suggest therefore that just as there are benefits even to a system that does not communicate with others to be able to reason about its own and others' actions, these benefits extend to what have traditionally been considered language understanding and use problems. A language processing system should be able to

- o plan utterances to achieve specific communicative goals, depending on its knowledge of the beliefs and intentions of its user, and
- o recognize the user's utterances as parts of larger plans that may be communicated over several utterances, or which the user intends to have inferred based on shared beliefs.

We therefore propose that versatility, discrimination, and helpfulness can be obtained from a language understanding system operating according to the following cycle:

1. Observe the uttering of a sentence.

2. Based on the sentence's mood, attribute the effect of that act to be a want of the user.

3. Using intended plan recognition and shared beliefs, infer, if possible, how the observed action(s) fits into a plan achieving a goal the user is expected to have. If a plan cannot be uniquely specified, create a system goal to discover the user's goal.

4. Create system goals for goals that user intended the system to achieve. A non-single-minded systems would have to decide which of the user's goals for the system should in fact become the system's goals.

5. Using private beliefs, determine obstacles at which the user's plan will fail, or where the user will need help.

6. Adopt the negation of some of those obstacles as goals for the system.

7. Using private beliefs, construct a plan achieving the system's goals, especially goals to overcome the user's obstacles. Depending on the goal, this plan may include communication actions, such as questions to clarify the user's goals.

8. Execute the resulting sequence (perhaps producing language).

9. Go to step 1.

We suggest that systems designed along these lines should be able to exhibit the intention recognition and helpful behavior necessary to solve the problems identified in the transcript fragments. In the following chapter, we give two examples of such systems and describe the problems they are equipped to handle.

## 5. DISCUSSION OF IMPLEMENTED SYSTEMS

Various parts of the general design outlined above have been implemented in two systems so far, operating in quite different domains. A system developed by Allen at the University of Toronto plays the role of an information clerk at a train station. It was tested on samples of actual dialogues collected at Union Station in Toronto [31]. The context of these dialogues is quite restricted but the linguistic behavior is nevertheless complex. A second system, implemented at Bolt, Beranek, and Newman (BBN), engages in dialogues about a display screen. Both systems distinguish the beliefs and wants of the user from their own,[19] and can recognize indirect speech acts. Allen's system can also analyze short sentence fragments, and provide helpful replies. We will give examples of the behavior of these systems and sketch their design.

The description of the systems given here is brief. The plan inference mechanism of Allen's thesis is described in [1], and the treatment of indirect speech acts in [45]. Implementation details can be found in [2]. The BBN system is

---

[19] Neither system has a logically complete inference mechanism to handle beliefs and wants. For steps in that direction see [33, 42].

described in [5, 58].

### 5.1 Allen's System.

Allen's system expects users to want to board or meet trains. In dialogue fragment D6, the system literally answers D6-1, but also provides gate information, which it deduces the user does not know but needs in order to achieve a goal he did not express.

D6-1 U: When does the Montreal train leave?
2 S: 3:15 at gate 7.

The system can also infer intentions based on sentence fragments. For example, to provide the reply D7-2 the system uses its expectations to infer that the user's goal is to board the 3:15 train to Windsor, and that he also needs the gate information to do so.

D7-1 U: The 3:15 train to Windsor?
2 S: Gate 10.

The fragment was analyzed using without reconstituting a syntactic analysis, as in LIFER [28].

In dialogue D8, the system must generate a question to disambiguate trains <u>to</u> Windsor and trains <u>from</u> Windsor.

D8-1 U: When is the Windsor train?

2 S: Do you want to go to Windsor?

3 U: yes

4 S: 3:15

The system correctly analyzes a wide range of indirect requests, including conventional ones such as:

18 Do you know when the Windsor train leaves?

19 I want you to tell me when the Windsor train leaves.

20 I want to know when ...

21 Tell me when ...

22 Can you tell me when ...

23 Will you tell me when ...

It can also handle non-conventional forms such as the following:

24 John asked me to ask you when the next train to Windsor leaves.

25 John wants to know when the next Windsor train leaves.

All these examples are handled by the same mechanism, a straightforward implementation of the cycle given in section 4, consisting of four major stages.

- o a parser, which uses syntactic and semantic information to produce a literal interpretation of the input, or a partial interpretation in the case of sentence fragments;

- o a plan recognition component that, given a set of expected high level goals (e.g. board a train, meet a train, ...) and an observed action (the parser output), infers a plan that links the two;
- o an obstacle detection component, which analyzes the plan produced above for steps that the user cannot perform (easily) without assistance from the system;
- o a plan construction component that, given a goal, plans a course of action that may involve communication (as in [16]).

Only the plan recognition and obstacle detection stages will be considered in more detail. The other components were implemented in order to create a complete system and used existing technology.

The system represents all the actions it can reason about, including the speech acts, in terms of three formulas (similar to the ones used in the STRIPS planning system [20]):

- o Preconditions -- Conditions necessary to the successful execution of the action.
- o Effects -- Conditions that become true as a result of the execution of the action.
- o Means -- Conditions that must be achieved during the execution of the action.

The parser produces an analysis of each input sentence in two parts: the function of the sentence is described in terms of a small number of actions corresponding to declaratives, interrogatives, and imperatives. The content of the sentence becomes an argument to the chosen act.

The plan recognition process can be viewed as a search through a space of pairs of plan fragments. One member of each pair is a partial plan inferred from the observed action by the application of plan recognition rules, and the other is a partial plan inferred from an expected goal by the application of plan construction rules. The plan construction and plan recognition rules are domain-independent and are inverses.

None of these rules (about 16) is logically valid, so they are used as "legal move generators" in a game where the positions are pairs of plan fragments. The positions are evaluated by a set of heuristics. At any time the highest rated pair is extended by the plan recognition or construction inferences.

Different sets of heuristics measure:

- o how well-formed the partial plans are in the given context;
- o how well the observed action fits with the expected goals; and
- o how likely it is that the inferences proposed were intended by the speaker.

We shall discuss some examples of each of these in turn.

An example of a heuristic from the first class is

> Decrease the rating of a partial plan in which the effects of a pending act already hold.

The degree of compatibility between a partial plan derived from the observed action and a partial plan derived from an expected goal is measured by how many common objects and relations are referenced by both. A heuristic from the second class favors plan pairs that have many common objects and relations.

The last class of heuristics deals with evaluating the likelihood that the speaker intended the inferences to be made, and contains two heuristics. The first heuristic was mentioned earlier in section 4. It favors expanding a partial plan that gives rise to a single line of inference over one that gives rise to many possible mutually exclusive inferences. The second one favors an inference that assumes that an agent wants his intentions to be recognized over one that does not. Thus, intended plan recognition is favored over keyhole plan recognition.

In summary, intended plan recognition only continues while there is a well-defined path to follow. If the system has a poor model of the user, then such well-defined paths will seldom occur and utterances will tend to be analyzed more literally. As the system's model of the user improves, its responses become more useful and less literal.

## 5.2 The BBN System

The BBN system uses Allen's model and engages in dialogues about a bit-map display screen that is under the system's control. It is intended as a prototype decision support system whose salient features include the use of both graphic and linguistic means of communication for both input and output. The system has a primitive capability to use shared beliefs to discriminate among user intentions. Its shared beliefs include the contents of the display screen, its display capacities, and expectations of conversation patterns. The system can display ATN grammars, change the scale of a display to simulate "zooming", and highlight entities on the screen. The system participated in the following dialogue.

D9-1 U: Show me the clause level network.
2 S: [system displays network on screen]
3 U: Show me S/NP.
4 S: [system highlights state S/NP]
5 U: Focus in on the preverbal constituents.
6 S: [system changes scale and display]
7 U: No. I want to be able to see S/AUX.
8 S: [system reduces scale so that state S/AUX is visible]

As an illustration of intent discrimination based on visual context, notice that although the two requests by the user in

D9-1 and D9-3 are of the same form, the system response differs based on what is on the screen. Since the screen is empty, the first request is interpreted to be a display operation. With the second, since what is asked for is already on the screen, the request is interpreted to be a for a highlighting operation rather than simply for a display of a large S/NP state.

As an illustration of intent discrimination based on shared expectations, notice that the BBN system analyzes "No" as a rejection, causing it to expect that the user will want to modify the display.[20] This is in contrast to PLANES´ ignoring "No" in D1-3. The remainder of D9-7 ("I want to be able to see S/AUX") is analyzed as communicating not just that the user wants the system to take note of his goals, but also that the user wants the system to plan and do something to satisfy them. The system arrives at two alternative plans the user might have in mind -- to erase the screen and then display S/AUX alone (analogous to PLANES´ analyzing D1-3 in isolation), or to include S/AUX into the current display. Since it is shared knowledge that the latter action is characterized as a display modification action, and since the previous rejection caused the system to expect the

[20] For an exploration of the use of other "clue words" see [47].

user to want to modify the display, the system infers that the user wanted it to recognize that he wanted it to include S/AUX. The system adopts that goal as its own and includes S/AUX into the display.

From an implementation point of view, one of the most important differences between Allen's system and the BBN system is that the latter is designed to systematically short-cut some of the inference chains necessary for indirect speech act interpretation (cf. Morgan's [43] "short-circuited implicatures"). For example, associated with the general action "User asks system whether system can do an action" there is a short-cut inference rule stating that, under certain conditions, the utterance should be interpreted as communicating the user's intention that the system do that action. Using such a rule, the system might respond to the utterance "Can you move it up?" (referring to an entity on the screen) with "yes" followed by a display action.[21]

The conditions governing a short-cut rule are derived from the chain of inferences that would be necessary to steer the more general plan recognition process to the same interpretation. The

---

[21] The example is due to Sidner and Israel [58].

appropriate conditions for the example rule include its being shared knowledge that the system has the capacity to move up entities on the screen, and, for non-single-minded systems, its not being shared knowledge that the system wants not to move entities (or that entity) up.

Importantly, the full plan recognition technique is still available for use, either after short-cut rules have been applied or when they have failed.[22] As an illustration of the latter case, consider the "Can you..." example. If the above rule were inapplicable (perhaps because the system's capacities were not shared knowledge), the full inference process would yield a literal interpretation as a question. Subsequent keyhole plan recognition might lead the system to respond "Yes", and to offer help by saying "Should I [move it]?"

Regarding the former case, some analyses involve the combined use of short-cuts and the general plan recognition mechanism. For example, a system asked "Can you find my recommendation letters?" has to reason first that it should actually find the letters, and then that it should show the

[22] This distinguishes our method from Brown's [6] and Lehnert's [34] whose rules are not embedded in a general reasoning mechanism.

letters to the user. Again, while this sequence could perhaps be short-cut, the possibility of reasoning about subsequent actions must always be considered.

Although the short-cut method may still be "less efficient" than ad hoc mappings, such as interpreting all "Can you do X?" questions as requests, it covers more cases. We believe that it is through rule compilation techniques like this that one should strive for systems that are both correct and efficient.

## 6. CONCLUDING REMARKS

Evidence has been presented here that users of question-answering systems expect them to do more than just answer isolated questions -- they expect systems to engage in conversation. In doing so, the system is expected to allow users to be less than meticulously literal in conveying their intentions, and it is expected to make linguistic and pragmatic use of the previous discourse.

Conversation systems should be designed to be goal-directed and helpful. To this end, we have proposed and illustrated a system architecture, based on reasoning about beliefs, goals, and actions. The system design is intended not only to extend the current versatility and discrimination of question-answering systems, but also to serve as a framework for developing natural language systems for applications requiring greater versatility, discrimination, and context-dependence. The more versatile the system, the more it will require the machinery proposed here.

Similar arguments can be made for modality requirements. Systems employing both linguistic and graphic means of communication will need a common framework for representing and reasoning about what is to be communicated, independent of modality. A system built along the lines proposed here, would have a range of communicative actions, some of which could employ

graphic means. In solving a problem, either (or even both) means would be used as requested or as helpfully appropriate.

This program of research should rest on a strong theoretical foundation. Consequently, research on formalisms for representing and reasoning about beliefs, desires, actions, and plans are crucially important. When applied to communicative actions, we expect such formalisms to lead to a formal theory of goal-oriented conversation.

Two examples of theoretical areas in which better formalisms would pay great dividends are worth noting. First, the STRIPS-like formalism used for the representation of actions in the two systems discussed here is insufficient for handling complex actions involving sequencing, conditionals, disjunctions, and parallelism, and is thus inadequate to express requests to do such acts. The formalism is also inadequate as Moore [42] points out to express what the agent of an action knows (and does not know) after the success or failure of an act. Moore's logic of knowledge and action offers solutions to some of these problems and is being applied to the planning of speech acts by Appelt [3].

Secondly, current algorithms do not adequately construct and recognize plans that achieve multiple goals. This appears to be one of the most fertile areas to pursue since it is well known

that utterances can simultaneously achieve goals of referring, focussing, and discourse structuring.

We conclude that question-answering interactions should be treated as degenerate cases of conversation. We propose that more general conversational capabilities be developed and applied to building question-answering systems as well as others of greater versatility. Some would claim that natural or quasi-natural language systems cannot and should not be competent conversants even in restricted domains [57], and hence such research should be abandoned. We contend, however, that not only is it proper for computational linguistics research to address problems of conversation directly, but that it is important to do so, and that modest progress toward attaining reasonable goals is currently being made. There is much work to do.

**REFERENCES**

1. Allen, J. F., & Perrault, C. R. Analyzing intention in dialogues. Artificial Intelligence 15, 3, 1980, 143-178.

2. Allen, J. F. A plan-based approach to speech act recognition. Technical Report 131, Department of Computer Science, University of Toronto, January, 1979.

3. Appelt, D. Problem-solving applied to language generation. Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics, Philadelphia, Pennsylvania, 1980.

4. Austin, J. L. How to do things with words. Oxford University Press, London, 1962.

5. Brachman, R., Bobrow, R., Cohen, P., Klovstad, J., Webber, B. L., & Woods, W. A. Research in natural language understanding. Technical Report 4274, Bolt Beranek and Newman Inc., August, 1979.

6. Brown, G. P. *Characterizing Indirect Speech Acts.* American Journal of Computational Linguistics 6, 3-4, 1980, 150-166.

7. Bruce, B. C., & Newman, D. Interacting plans. Cognitive Science 2, 1978, 195-233.

8. Bruce, B. C. Belief systems and language understanding. Technical Report 2973, Bolt Beranek and Newman Inc., 1975.

9. Bruce, B. C. Generation as a Social Action. Proceedings of the conference on theoretical issues in natural language processing, Cambridge, Mass., June, 1975, 64-67.

10. Bruce, B. C. Natural communication between person and computer. In Strategies for Natural Language Processing, Lehnert, W., & Ringle, M., (Eds.), Hillsdale, N. J., 1981.

11. Carbonell, J. G. Jr. POLITICS: Automated ideological reasoning. Cognitive Science 2, 1978, 27-51.

12. Clark, H. H., & Marshall, C. Definite reference and mutual knowledge. In Elements of Discourse Understanding, Joshi, A. K., Sag, I. A., & Webber, B., (Eds.), New York, 1981.

13. Codd, E. F., Arnold, R. S., Cadiou, J. M., Chang, C. L., & Roussopoulos, N. RENDEZVOUS VERSION 1: An experimental English-language query formulation system for casual users of relations data bases. Research Report RJ2144, IBM Research Laboratory, January, 1978.

14. Codd, E. F. Seven steps to RENDEZVOUS with the casual user. Proceedings IFIP TC-2 Working Conference on Data Base Management Systems, North-Holland Publishing Co., Amsterdam, 1974, 179-200.

15. Cohen, P. R., & Perrault, C. R. Elements of a plan-based theory of speech acts. Cognitive Science 3, 1979, 177-212.

16. Cohen, P. R. On knowing what to say: Planning speech acts. Ph.D. Th., University of Toronto, Toronto, January 1978. Technical Report No. 118, Dept. of Computer Science.

17. Cohen, P. R. The need for identification as a planned action. Unpublished manuscript

18. Dennett, D. Brainstorms: Philosophical essays on mind and psychology. Bradford Books, 1978.

19. Deutsch, B. G. The structure of task-oriented dialogues. In Proceedings of the IEEE symposium on speech recognition, L. D. Erman, (Ed.), Pittsburgh, Pennsylvania, 1974.

20. Fikes, R., & Nilsson, N. J. STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intelligence 2, 1971, 189-208.

21. Genesereth, M. R. Automated consultation for complex computer systems. Ph.D. Th., Department of Computer Science, Division of Applied Sciences, Harvard University, September 1978.

22. Grice, H. P. Meaning. Philosophical Review 66, 1957, 377-388.

23. Grosz, B. J. The representation and use of focus in dialogue understanding. Technical Report 151, Artificial Intelligence Center, SRI International, July, 1977.

24. Grosz, B. J. Discourse Analysis. In Understanding Spoken Language, D. Walker, (Ed.), Chap. IX, New York City, 1978, 235-268.

25. Grosz, B. J. Utterance and objective: Issues in natural language communication. Proceedings of the Sixth International Joint Conference on Artificial Intelligence, Tokyo, 1979, 1067-1076.Reprinted in AI Magazine, American Assocation for Artificial Intelligence, 1, 1, Spring, 1980.

26. Harris, R. User oriented data base query with the ROBOT natural language query system. International Journal of Man-Machine Studies 9, 1977, 697-713.

27. Hendrix, G. G., Sacerdoti, E. D., Sagalowicz, D., & Slocum, J. Developing a natural language interface to complex data. ACM Transactions on Database Systems 3, 2, 1978, .

28. Hendrix, G. G. LIFER: A natural language interface facility. Technical note 135, Artificial Intelligence Center, SRI International, 1976.

29. Hendrix, G. G. Future prospects for computational linguistics. Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics, Philadelphia, June, 1980, 131-135.

30. Hobbs, J., & Evans, D. Conversation as planned behavior. Cognitive Science 4, 4, 1980, 349-377.

31. Horrigan, M. K. Modelling simple dialogues. Technical Report 108, Department of Computer Science, University of Toronto, 1977.

32. Kaplan, S. J. Indirect responses to loaded questions. Proceedings of the Second Conference on Theoretical Issues in Natural Language Processing, Urbana, Illinois, 1978.

33. Konolige, K. A first-order formalization of knowledge and action for a multiagent planning system. Technical note 232, Artificial Intelligence Center, SRI International, December, 1980. (to appear in Machine Intelligence, Volume 10).

34. Lehnert, W. G. The process of question-answering -- A computer simulation of cognition. Lawrence Erlbaum Associates, Hillsdale, N. J., 1978.

35. Levin, J. A., & Moore, J. A. Dialogue games: Metacommunication structures for natural language interaction. Cognitive Science 1, 4, 1977, 395-420.

36. Lewis, D. K. Convention: A philosophical study. Harvard University Press, Cambridge, Massachusetts, 1969.

37. Malhotra, A. Design criteria for a knowledge-based English language system for management: An experimental analysis. Ph.D. Th., Massachusetts Institute of Technology, MAC TR-146, 1975.

38. Mann, W., Moore, J., & Levin, J. A comprehension model for human dialogue. Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Cambridge, Massachusetts, August, 1977.

39. Mann, W. C. Why things are so bad for the computer-naive user. Technical Report RR-75-32, University of Southern California, Information Sciences Institute, March, 1975.

40. McCarthy, J., & Hayes, P. J. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In Machine intelligence 4, B. Meltzer & D. Michie, (Eds.), New York, 1969.

41. McCarthy, J. Ascribing mental qualities to machines. Technical Report STAN-CS-79-725, Computer Science Department, Stanford University, March, 1979.

42. Moore, R. C. Reasoning about knowledge and action. Technical Note 191, Artificial Intelligence Center, SRI International, October, 1980.

43. Morgan, J. L. Two types of convention in indirect speech acts. In Syntax and Semantics, Volume 9: Pragmatics, P. Cole, (Ed.), New York, 1978, 261-280.

44. Newell, A., & Simon, H. A. GPS: A program that simulates human thought. In Computers and thought, E. A. Feigenbaum & J. Feldman, (Eds.), New York, 1963.

45. Perrault, C. R., & Allen, J. F. A plan-based analysis of indirect speech acts. American Journal of Computational Linguistics 6, 3, 1980, 167-182.

46. Perrault, C. R., & Cohen, P. R. It's for your own good: A note on inaccurate reference. In Elements of Discourse Understanding, Joshi, A., Sag, I., & Webber, B., (Eds.), Cambridge, Mass., 1981.

47. Reichman, R. Plain-speaking: A theory and grammar of spontaneous discourse. Ph.D. Th., Department of Computer Science, Harvard University, Cambridge, Massachusetts, 1981.

48. Robinson, A. E., Appelt, D. E., Grosz, B. J., Hendrix, G. G., & Robinson, J. J. Interpreting natural-language utterances in dialogs about tasks. Technical Note 210, Artificial Intelligence Center, SRI International, March, 1980.

49. Sacerdoti, E. D. Language access to distributed data with error recovery. Proceedings of the Fifth International Conference on Artificial Intelligence, Cambridge, Massachusetts, 1977, 196-202.

50. Schank, R., & Abelson, R. Scripts, plans, goals, and understanding. Lawrence Erlbaum Associates, Hillsdale, N. J., 1977.

51. Schiffer, S. Meaning. Oxford University Press, London, 1972.

52. Schmidt, D. F., Sridharan, N.S., & Goodson, J. L. The plan recognition problem: An intersection of artificial intelligence and psychology. Artificial Intelligence 10, 1979, 45-83.

53. Schmidt, C. F. Understanding human action. Proceedings of Conference on Theoretical Issues in Natural Language Processing, Cambridge, Massachusetts, 1975.

54. Schmidt, C. F. Understanding human action: Recognizing the plans and motives of other persons. In Cognition and Social Behavior, Carrol and Payne, (Eds.), 1976, 47-67.

55. Searle, J. R. Speech acts: An essay in the philosophy of language. Cambridge University Press, Cambridge, 1969.

56. Searle, J. R. A taxonomy of illocutionary acts. In Language mind and knowledge, K. Gunderson, (Ed.), 1976.

57. Shneiderman, B. Natural vs. precise concise languages for human operation of computers: Research issues and experimental approaches. Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics, Philadelphia, Pennsylvania, June, 1980, 139-141.

58. Sidner, C., & Israel, D. Recognizing intended meaning and speaker's plans. Unpublished manuscript, Bolt Beranek and Newman

59. Sidner, C. L. Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse. Technical Report 537, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, June, 1979. PhD. Thesis.

60. Sidner, C. L. Protocols of users manipulating visually presented information with natural language. Unpublished manuscript, Bolt Beranek and Newman, May, 1980

61. Tennant, H. Evaluation of natural language processors. Technical Report T-103, Coordinated Science Laboratory, University of Illinois, November, 1980. Ph. D. Thesis.

62. Thompson, F. B., & Thompson, B. H. Practical natural language processing: The REL system prototype. In Advances in Computers, Rubinoff, M., & Yovits, M. C., (Eds.), New York, 1975, 109-168.

63. Waltz, D. L. An English language question answering system for a large relational database. Communications of the ACM 21, 7, 1978, 526-539.

64. Webber, B. L. A formal approach to discourse anaphora. BBN Report 3761, Bolt Beranek and Newman, May, 1978.

65. Wilensky, R. Understanding goal-based stories. Ph.D. Th., Department of Computer Science, Yale University, New Haven, Connecticut, 1978. Research Report No. 140.

66. Winograd, T. Understanding Natural Language. Academic Press, New York, 1972.

67. Woods, W., Bates, M., Brown, G., Bruce, B., Cook, C., Klovstad, J., Makhoul, J., Nash-Webber, B., Schwartz, R., Wolf, J., & Zue, V. Speech Understanding Systems -- Final Technical Progress Report. Technical Report 3438, Bolt Beranek and Newman Inc., 1976.

68. Woods, W. A., Kaplan, R. M., & Nash-Webber, B. The lunar sciences natural language system. Technical Report 2378, Bolt Beranek and Newman Inc., 1972.

Official Distribution List

Contract N00014-77-C-0378

| | Copies |
|---|---|
| Defense Documentation Center<br>Cameron Station<br>Alexandria, VA 22314 | 12 |
| Office of Naval Research<br>Information Systems Program, Code 437<br>Arlington, VA 22217 | 2 |
| Office of Naval Research<br>Code 200<br>Arlington, VA 22217 | 1 |
| Office of Naval Research<br>Code 455<br>Arlington, VA 22217 | 1 |
| Office of Naval Research<br>Code 458<br>Arlington, VA 22217 | 1 |
| Office of Naval Research<br>Branch Office, Boston<br>495 Summer Street<br>Boston, MA 02210 | 1 |
| Office of Naval Research<br>Branch Office, Chicago<br>536 South Clark Street<br>Chicago, IL 60605 | 1 |
| Office of Naval Research<br>Branch Office, Pasadena<br>1030 East Green Street<br>Pasadena, CA 91106 | 1 |
| Dr. Martin Epstein<br>National Library of Medicine<br>Bldg. 38A, 8th Floor Lab.<br>8600 Rockville Pike<br>Bethesda, MD 20209 | 1 |
| Naval Research Laboratory<br>Technical Information Division<br>Code 2627<br>Washington, DC 20380 | 6 |

Naval Ocean Systems Center 1
Advanced Software Technology Division
Code 5200
San Diego, CA 02152

Dr. A.L. Slafkosky 1
Scientific Advisor
Commandant of the Marine Corps (Code RD-1)
Washington, D.C. 20380

Mr. E. H. Gleissner 1
Naval Ship Research & Development Center
Computation & Mathematics Dept.
Bethesda, MD 20084

Captain Grace M. Hopper 1
NAICOM/MIS Planning Branch (OP-916D)
Office of Chief of Naval Operations
Washington, D.C. 20350

Mr. Kin B. Thompson 1
NAVDAC 33
Washington Navy Yard
Washington, D.C. 20374

Advanced Research Projects Agency 1
Information Processing Techniques
1400 Wilson Boulevard
Arlington, VA 22209

Captain Richard L. Martin, USN 1
3311 W. Avenue
Newport News, VA 23607

Director, National Security Agency 1
Attn: R54: Mr. Page
Fort G.G. Meade, MD 20755

Director, National Security Agency 1
Attn: R53, Mr. Glick
Fort G.G. Meade, MD 20755

Major James R. Kreer, Chief 1
Information Sciences
Dept. of the Air Force
Air Force Office of Scientific Research
European Office of Aerospace Research
and Development
Box 14
FPO New York 09510